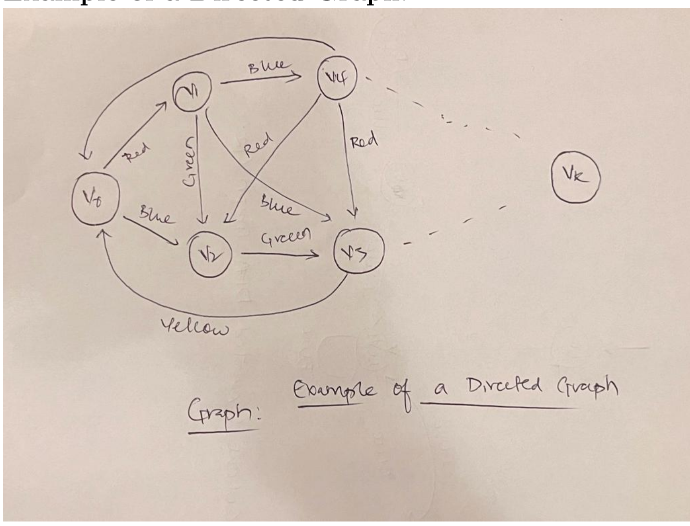
CptS 515 ADVANCED ALGORITHMS MIDTERM

Sharath Kumar Karnati - 011852253

October 2023

Let C be a finite set of colors and G be a directed graph where v0 is a designated start node. Each edge of G is labeled with a color in C and multiple edges can share the same color. From v0, one may have infinitely many walks since there is no upper bound on the length of walks and there could be cycles in G. For each such walk (that must start with v0), we may collect the sequence c() of colors on the edges on the walk. We use P to denote the set of all walks starting from v0 and use C to denote all the resulting color sequences: C = c(α) : $\alpha \in$ P . Clearly, C may be an infinite set (of color sequences).

PROBLEM DESCRIPTION:
Example of a Directed Graph:



Given,
Labeled Graph : G
Finite colour set: C
Walk or path starting from V_0 : α
Each edge has a colour that belongs to C
Sequence of colors on the edges of the walk : C(α)
Set of all walks α : P
M1: Metric which is a function of G that measures Non determinism on all

walks in P.

Deterministic Choices:

Deterministic choices are those in which there is only one possible path or option to follow from a given state or point. In deterministic choices, the transition or decision made at a node is entirely predictable based on the current state and input.These choices are typically associated with deterministic algorithms and finite automata.

Examples of Deterministic choices : Deterministic Finite Automata (DFA) , and Deterministic algorithms like Binary Search etc.

Non-Deterministic Choices:

Non-Detrministic choices are those that refer to situations in which there are multiple possible paths or options to follow from a given state or node. The choice made at a node is entirely predictable and have multiple outcomes are possible. These choices are often associated with non-deterministic algorithms and non-deterministic automata.

Examples of Non-deterministic choices: Non-Deterministic Finite Automata(NFA) and Non-deterministic algorithms like Travelling salesman problem using brute force.

QUESTION 1:

(10pts) Let u be a node in G. From this u, one may have multiple outgoing edges, say ⟨u, v1⟩, ⟨u, v2⟩, · · ·, ⟨u, vk⟩, for some $k2$, whose colors are all the same. One can understand the same color as a triggering event that leads from current node u to a "next" node chosen nodeterministically from v1, · · · , vk. Clearly, if I show you a walk in P , then there could be many edges on the walk that are the result of many nondeterministic choices. Please develop and justify a metric M1 which is a function of G that measures the nondeterminism on all walks in P . (Your M1 is high when nondeterminism is high.) You need also show me an algorithm in computing such M1. In case when efficient algorithm is hard to obtain, please also provide an approximation algorithm.

SOLUTION :

For this problem, we are asked to define a metric(M1) that measures the Non-deterministic choices on all walks from P starting from V_0

Given, A labeled Graph G, each edge has a colour $\in C$

TO FIND : A metric M_1 which is a number $\in R^+$ (non negative)

METRIC : A metric is a quantifiable and measurable variable or a function that is used as a unit for assesment of a problem. These are quantitative measures used to evaluate the performance, efficiency, and effectiveness for an algorithm.

To DO: To design an algorithm to compute a metric that helps us with a non deterministic directed graph.

From the given,

u = initaial node in Graph G.

From u we have multiple outgoing ways.

$< u, V1 >, < u, V2 >, \ldots\ldots\ldots < u, Vk >$ for some $k >= 2$.

From the given lets create a graph and lets consider this to solve this problem.

For this we can say that the metric for the non determinism will be higher when we have multiple choices. We are gonna use metric M1 to measure the non determinism of each walk in P.

ALGORITHM :

- 1. Let us consider this graph G starting from u as a initial node or starting node, as mentioned in the question.

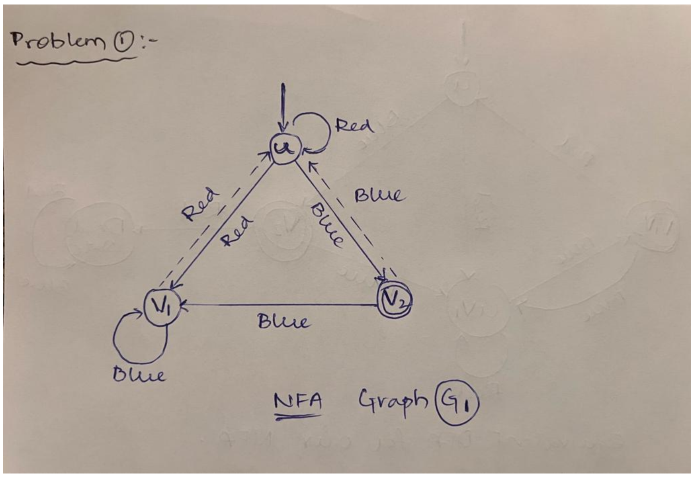
- 2. Now, we are gonna consider G as a Non deterministic finite automata(NFA).

- 3. Generally , NFA is defined by (Q, Σ , q_0, F,A, δ)
where,
Q is the set that contains all states.(A finite set of states)
Σ is a finite set of input that NFA can read.
q_0 is a initial start state that belongs to Q.
F is the set of accept states, A set of states from the set Q, which are the accept states. If NFA reaches one of these states after processing the input, then that input is considered.
A is the acceptance.
Transition Function (δ) : A function that maps a state and an input symbol to a set of states formally, δ:$Q \times \Sigma \rightarrow 2^Q$, this represents the power set of Q. This non deterministic aspect allows an NFA to have multiple possible next states for a given input, giving it the ability to guess or explore different paths. It basically says number of total possibilities.

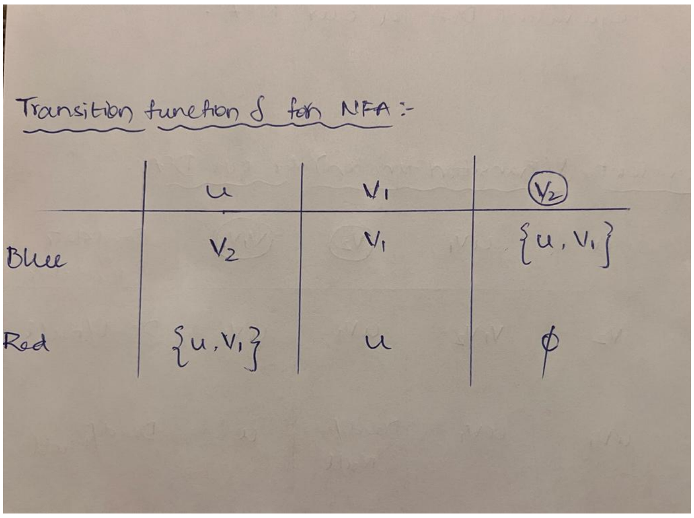
- 4. In our case,
Q is the set of all nodes in G.
Σ is the set of all colors in G.

- 5. Now, lets consider this simple NFA for an instance to determine the metric.

  Example Graph: G1(NFA):

- **6. Transition function for our NFA is :**

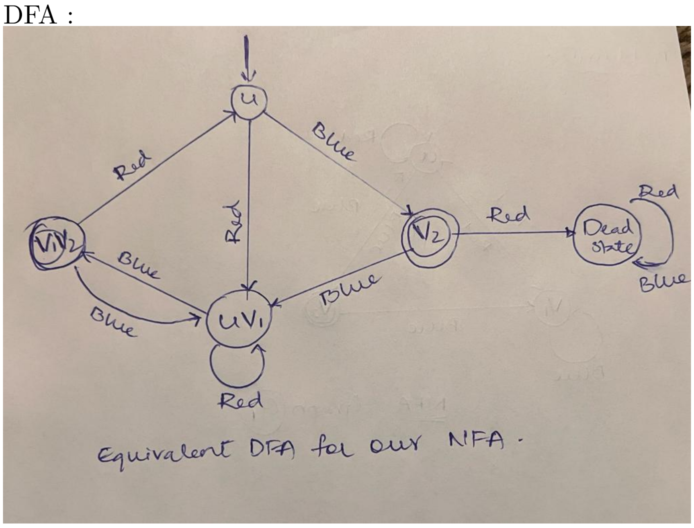


Transition function δ for NFA :-

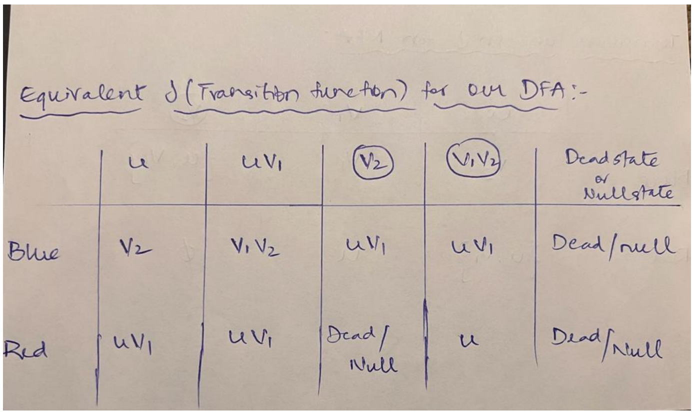
	u	V_1	V_2
Blue	V_2	V_1	$\{u, V_1\}$
Red	$\{u, V_1\}$	u	ϕ

- 7. We know that, for an every existing Non- Deterministic finite Automata (NFA) there should be a existing equivalent Deterministic finite Automata (DFA) . Now , lets convert our NFA into a equivalent DFA.

  DFA :



- **8. Transition function for our DFA is :**


Equivalent δ (Transition function) for our DFA:-

	u	uV_1	V_2	V_1V_2	Dead state or Null state
Blue	V_2	V_1V_2	uV_1	uV_1	Dead/null
Red	uV_1	uV_1	Dead/Null	u	Dead/Null

- 9. Here, we have done mapping of NFA to get the mapping of its equivalent DFA. We have constructed the DFA with the help of this.
- 10. In this DFA,
  u - Node is start point.
  D - Dead state or null state.
- 11. Each node in the DFA is a collection of every state that the previous state might reached by using the same colour.
- 12. For a sequence C, the combination of all the metrics that it went through will be its metric.
  13. From the Shannon Entropy , we can state that " - $\Sigma_i \mathrm{p}_i\ log_2\ p_i$ " as a metric M1, that can measure the non determinism and is a part of a function of G.
  We use log sum as it generally produces metrics in the range of 0 and 1 where, 1 being a entirely non deterministic and 0 being a strictly deterministic.

EXPLANATION : Here we used shanon entropy to define the metric M1 . Shanon entropy is typically used for random variables, which are variables whose values are determined by chance,for example: when flipping a coin the outcome is a random variable . It is used to measure the average uncertainty or information content associated with a random variable.If all the outcomes are equally likely, the entropy is at its maximum because there is a high degree of uncertainty or surprise. If one outcome is very likely and the other are unlikely, the entropy is lower because there is less surprise associated with the variable. When can consider non determinism and determinism as a random variable as there always be an uncertainty associated with it. Shanon entropy is associated with log sum hence we have a certain range of possible outcome 1 and 0 , where 1 is taken as entirely non-deterministic while 0 is taken as strictly deterministic. We have defined a metric with the help of entropy which can be used to measure the non determinism of the each walk in P for the labelled graph G.

QUESTION : 2 (10pts) Multiple walks can share the same color sequence. Hence, if we measure nondeterminism from the angle of color sequences in C, the metric would be very different. Color sequences in C may also have "nondeterminism" which can be understood in the following way. Consider two color sequences in C, say red, yellow, green, green, red, ... and red, yellow, blue, green, yellow.... The first two colors in the sequences are the same: red followed by yellow. However, the third colors are different (green in the first sequence and blue 1 in the second). One may say that there is a nondeterministic choice of the next color right after the first two colors. Please develop and justify a metric M2 which is a function of G that measures the nondeterminism on all color sequences in C. You need also show me an algorithm in computing such M2. In case when efficient algorithm is hard to obtain, please also provide an approximation algorithm.

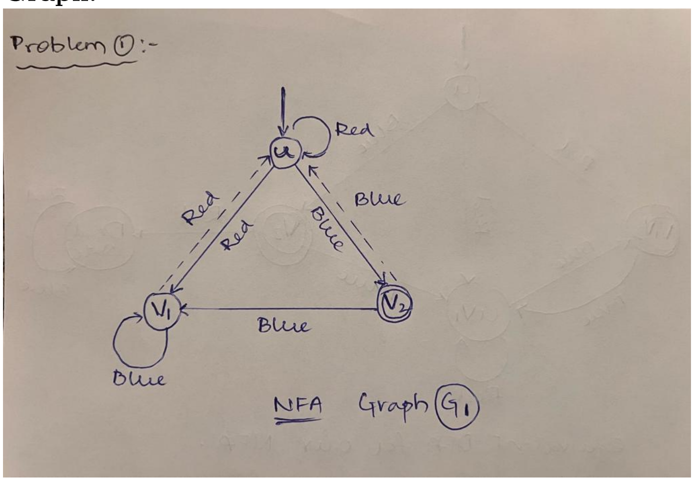
SOLUTION:

For this problem, we are going to consider non determinism for the multiple color choices that are made. We can observe that some nodes have 2 distant outgoing colour paths or data. Therefore, we can state that the more the colour options the higher the non determinism will be.

From the question,
Given,
C: Set of finite colors.
G: Labeled graph
To Find: Metric M2, which is the function of G that measures non determinism of all colour sequences.

In developing the algorithm, we are gonna consider the same graph that was mentioned in the last problem and from the we are gonna define the metric M2 that defines the non determinism for the color sequences.

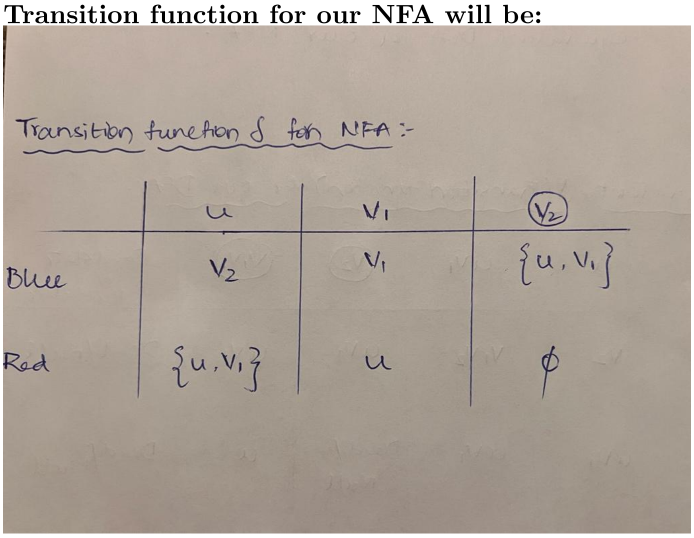
Graph:



ALGORITHM:

- 1. Firstly , lets consider the above graph G as a Non-deterministic finite Automata (NFA).
- 2. We know that generally NFA is defined by (Q, Σ , q_0, F,A, δ)
  where,
  Q is the set that contains all states.(A finite set of states)
  Σ is a finite set of input that NFA can read.
  q_0 is a initial start state that belongs to Q.
  F is the set of accept states, A set of states from the set Q, which are

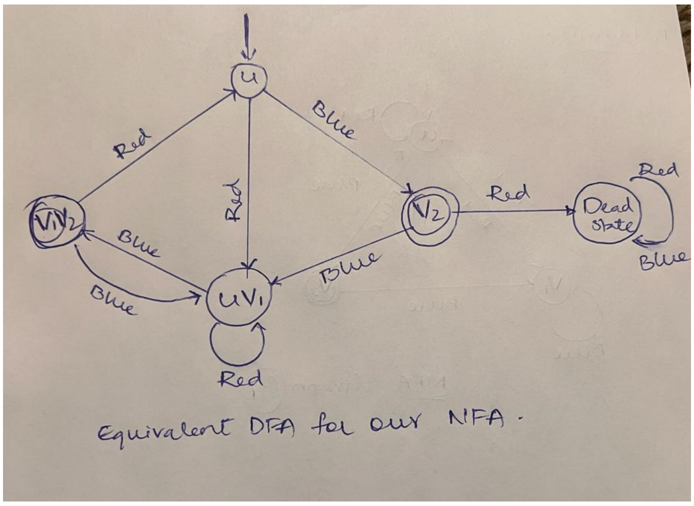
the accept states. If NFA reaches one of these states after processing the input, then that input is considered.
A is the acceptance.
Transition Function (δ) : A function that maps a state and an input symbol to a set of states formally, $\delta{:}Q{\times}\Sigma \rightarrow 2^{Q}$, this represents the power set of Q. This non deterministic aspect allows an NFA to have multiple possible next states for a given input, giving it the ability to guess or explore different paths. It basically says number of total possibilities.
Transition function for our NFA will be:



Transition function δ for NFA :-

	u	V_1	V_2
Blue	V_2	V_1	$\{u, V_1\}$
Red	$\{u, V_1\}$	u	ϕ

- 3. For this problem,
  Q is the set of all nodes in G.
  Sigma is the set of all colors in G.
- 4. Now lets consider the above NFA as an instance.
- 5.Now, we know that there always be an equivalent DFA for every NFA.
- 6. Lets convert our NFA into its equivalent DFA
- 7. We are doing mapping of the NFA to find its equivalent DFA.We have constructed the DFA with the help of this.
  DFA :

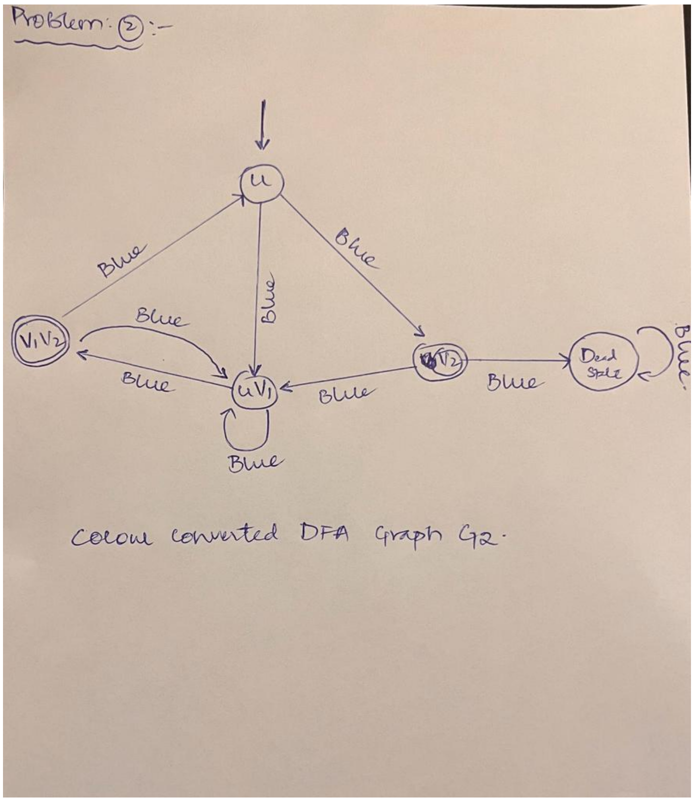


Transition function for our DFA :

Equivalent δ (Transition function) for our DFA:-

	u	uV_1	V_2	V_1V_2	Dead state or Null state
Blue	V_2	V_1V_2	uV_1	uV_1	Dead/null
Red	uV_1	uV_1	Dead/Null	u	Dead/Null

- 8.In this DFA,
  U : Node is start point.
  D : Dead state or null state.
- 9. Now lets change all the edges colour into a same colour, Lets assume that the colour is blue.
  Converted DFA:(GRAPH G2):



- 10. In DFA, every node is nothing but the collection of all possible states that its previous node has reached by using the same colour.

- 11. Now, the metric M2 can be formulated as " $-\Sigma_i p_i \ log_2 \ p_i$". We can observe that it is same as Metric M1 but metric M2 is for DFA of different graph G2. As the metric is nothing but combination of all states that this has travelled through.We have used Shannon Entropy to define the metric M2.

- 12. Metric M2 helps us to calculate the non determinism of colour sequence. We have used log sum as it produces metrics in the range of 0 and 1 where, 1 being a entirely non deterministic and 0 being a strictly deterministic.

EXPLANATION : Here we used shanon entropy to define the metric M1 . Shanon entropy is typically used for random variables, which are variables whose values are determined by chance,for example: when flipping a coin the outcome is a random variable . It is used to measure the average uncertainty or information content associated with a random variable.If all the outcomes are equally likely, the entropy is at its maximum because there is a high degree of uncertainty or surprise. If one outcome is very likely and the other are unlikely, the entropy is lower because there is less surprise associated with the variable. When can consider non determinism and determinism as a random variable as there always be an uncertainty associated with it. Shanon entropy is associated with log sum hence we have a certain range of possible outcome 1 and 0 , where

1 is taken as entirely non-deterministic while 0 is taken as strictly deterministic. We have defined a metric with the help of entropy which can be used to measure the non determinism of the colour sequence for the labelled graph G. Here we can observe that we have same metrics for M1 and M2 however, they are defined for two different criteria's and two different graphs.

QUESTION 3:

In good old days, a program is understood as a functional unit so that testing refers to figuring out its input/output relation. However, nowadays, a program (such as embedded systems code) is often reactive: it constantly interacts with its environment so that when you test it, it will not just give you one pair of input/output; instead, it will give you a test; i.e., a sequence of pairs of input/output: (i1, o1), · · · , (in, on) for some n. Notice that the input sequence i1, · · · , in is called a test case (provided by a test engineer) and the output sequence o1, · · · , on is the test result. Now, we assume that each of ij and oj is a color in C. The program can be nondeterministic; i.e., one test case may have multiple test results. We now assume that the program is a blackbox (whose code is not available; not even assembly nor machine code nor design nor requirements. Do not use any code analysis techniques, design analysis techniques, requirements analysis techniques here since they are not applicable and you will get zero.). Sadly, researchers know little about testing a nondeterministic blackbox program! Good news is one can use the most stupid approach in testing such a program: try many many test cases and then collect many many test results. Our experiences are very intuitive: a more highly nondeterministic program is also harder to test.

(3.1, 10pts) Please develop and justify an algorithm to estimate, from those many many tests, how high the nondeterminism of the blackbox pro- gram under test is.

SOLUTION :

Given,
Sequence of pairs of input/output:
$(i_1, o_1), (i_2, o_2) \ldots\ldots (i_n, o_n)$
Input Sequence : $i_1, \ldots.., i_n$ (test case)
Output sequence : $o_1, \ldots\ldots., o_n$ (test result)
Lets assume that , i_j and o_j are color in C.

To find: Need to develop and justify an algorithm to measure how high the non determination of the blackbox program under test case. Black Box program : Black box program is a code where the source code is not available. Here we don't have any access to the source code. We even don't know how it functionally works. It's hard to predict it and has uncertain functionality.

For example: Lets consider working of a washing machine as our black box program under the test case.lets consider two input and output pairs(i_1, o_1) and (i_2, o_2). In the first case, the input(i_1) is "beep sound" and the output(o_1) is "lid is closed". In the second scenario, the input(i_2) is " beep sound", the output(o_2) is "Overload". We can observe that in the both cases we have same input but the output differed this behaviour is termed as non-determinism. And we can't predict this .It is always uncertain that we can't predict the output any ways. So, to determine the metric to measure the non determinism we are gonna develop a algorithm with the help of Shannon Entropy.

We are gonna use Shannon Entropy to predict the non determinism of the black box code.
ALGORITHM : To measure the non determinism for a black box code under test case mentioned.
Input: A test sequence of pairs of input/output: $(i_1, o_1), (i_2, o_2) \ldots\ldots (i_n, o_n)$ And n, which is a length of the test sequence.
Output : A metric to determine the non determinism for a black box code.
START :
1.Firstly, let's consider the test sequence given $(i_1, o_1), (i_2, o_2) \ldots\ldots (i_n, o_n)$ and n which is a length of test sequence.
2.We are gonna consider the total number of runs as T_n and we already know the length of sequence(n).Therefore, $T_n = (i_t 1, o_t 1), \ldots\ldots\ldots (i_t n, o_t n)$.
3.Now, for comparing lets setup a baseline.
4.Now, lets set T_1 as our baseline term and we will compare it with T_2 to determine the non-determinism of node, where T_1 and T_2 differs for the first time only.
5.Now, we will repeat the comparision and compare all the test sequences with T_1 (we should just neglect the comparison of T_1 with T_1).
6.If we observe the test sequences for input $i_t 2$, we will see that the output may have numerous choices to choose.
7.Now, to calculate it we are gonna use Shannon Entropy H(x)= -Σ_ip$_i$ log_2 p_i,where p_i is the probabilistic function which determines whether the node is a deterministic or not.
8.The entropy will provide only values which only ranges from 0 and 1. If it is a 0, then it is deterministic and if it is 1 then it is purely non deterministic.
9.We can also calculate the average of all entropy by using the formulae H(X)-Σ $(|X_v|/|X|)$ $\times$H(X$_v$).
10.This will be our non deterministic metric for the black box program under the given test case.
END

(3.2, 50pts) Please write a mini-paper (2-3 pages) to show an application and how your algorithm in 3.1 can be used, including but not limited to its strength and weakness. Your solutiosn to all of the problems are graded based on originality, depth of thinking, and quality of writing. When you use others ideas, please cite properly. As always, please use LATEX to prepare your turn-in.

SOLUTION :

For this I am gonna submit the paper separately and will attach to this file in canvas and upload this.

CptS 515 Advanced Algorithms MIDTERM

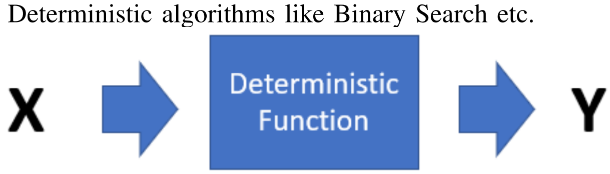
Sharath Kumar Karnati

Question:3 . (3.2, 50pts) Please write a mini-paper (2-3 pages) to show an application and how your algorithm in 3.1 can be used, including but not limited to its strength and weakness.

***Abstract*—In this paper we are gonna discuss briefly about the algorithm that we designed for the problem 3.1 which is generally used to define the metric that measures how high the non-determinism is for a black-box program. We are gonna talk about the application where we can use this algorithm and discuss briefly about it. In this paper we are gonna state the advantages and disadvantages that are associated with the use of the algorithm that we defined for the previous problem.**

I. Introduction :

Determinism: Determinism is in which there is only one possible path or option to follow from a given state or point. In deterministic choices, the transition or decision made at a node is entirely predictable based on the current state and input.In determinism there will be definitely only one input and one output. So, it will be easy to design an algorithm for these type of choices.These choices are typically associated with deterministic algorithms and finite automata. Examples of Deterministic choices : Deterministic Finite Automata (DFA) , and Deterministic algorithms like Binary Search etc.



Non-Determinism: Non-Determinism is which refers to situations in which there are multiple possible paths or options to follow from a given state or node. The choice made at a node is entirely predictable and have multiple outcomes are possible.Here we will have only one input but we will have multiple choices of output. SO, it always difficult to design a algorithm that measures non-determinism. These choices are often associated with non-deterministic algorithms and non-deterministic automata. Examples of Non-deterministic choices: Non-Deterministic Finite Automata(NFA) and Non-deterministic algorithms like Travelling salesman problem using brute force.

Black-Box Program: Black box program is a code where the source code is not available. Here we don't have any access to the source code. We even don't know how it functionally works. It's hard to predict it and has uncertain functionality.

We have used Shannon Entropy to find that metric that measures the non determinism of a black-box program.

Shannon entropy is typically used for random variables, which are variables whose values are determined by chance,for example: when flipping a coin the outcome is a random variable . It is used to measure the average uncertainty or information content associated with a random variable.If all the outcomes are equally likely, the entropy is at its maximum because there is a high degree of uncertainty or surprise. If one outcome is very likely and the other are unlikely, the entropy is lower because there is less surprise associated with the variable. When can consider non determinism and determinism as a random variable as there always be an uncertainty associated with it. Shannon entropy is associated with log sum hence we have a certain range of possible outcome 1 and 0 , where 1 is taken as entirely non-deterministic while 0 is taken as strictly deterministic.

Here we are gonna consider about a Real life(doctor-patient) scenario where our designed algorithm for black box program can be applied.

II. My Application :

According to the given problem, we have defined a algorithm for black-box program under a test sequence. Hence to develop a application that abides by the algorithm we should have a test sequence like (i1,o1),(i2,o2). . . . . . .(in,on), where i is input and o is the output for test sequence of length n. We know that as the walk is dependent on the prior data, so the historical input will definitely affect the black-box. As the problem is based on non determinism the black-box program should have many output choice. I am going to use decision analytical modelling in the scenario of a patient with a disease consulting a doctor.

Decision Analytical modeling: It is a valuable approach to decide informed decisions in situation where outcomes are uncertain and not fully predictable. This modeling allows us to incorporate probabilities, scenarios and sensitivity analysis into our decision-making process. This method is very helpful when we have uncertainty of outputs under certain provided conditions.

A. *Application Scenario:*

Lets us assume that there is person(X) who is suffering from a disease where his chances of cure is just 50 percent. There are 2 doctors A and B, who are both professionals but have a different success ratio. Let us assume that doctor A has a success ratio of 65 percent for the curing of the disease that X is suffering, while doctor B has a success ratio of 60 percent for the same.Here the person has only one (way) option to do that is going and consulting a doctor. The non determinism arises where he got an option to choose among the two doctors A and B(There can be even more options but for this case lets consider only two so that it would be easy to discuss) . Now , we will consider this whole scenario as a decision problem which often revolves around 2 or greater than two choices. From the Diagram(graph), the first step of non determinism arises when a person(X) has to choose between the two doctors (A and B) for the treatment of his disease. From this point the choices will even become longer depending on the doctor he chooses, we can see that he has 3 options if he chooses doctor A and has 2 options if he chooses doctor B. We can observe that the path is historic dependent as we can see that everything is related to the choice he made prior. From this we can calculate the metric to measure the non determinism using Entropy which will be a approximate or exact value. Here these decision trees are made greedily from the choices we made and runs until the criteria is reached. Here we have applied a stochastic approach in this case.

III. ALGORITHM IMPLEMENTATION :

- 1.Firstly, let's consider the test sequence given $X(i1, o1), (i2, o2) \ldots\ldots (in, on)$ And n which is a length of test sequence. And y is the total number of test runs. Here X may have various data-sets but their dimensions should be same matching to the length of test sequence as it is mentioned in the problem.
- 2.Now, from the amount of stationary points, lets sort the order in a way so that the first data-set should have the least amount of stationary points.
- 3.Now, lets consider 2 decision trees as a black-box and we will set their iteration times with T.
- 4.Now lets consider two sets $X_a = (i_a1, o_a1), \ldots\ldots (i_an, o_an)$ and $X_b = (i_b1, o_b1), \ldots\ldots (i_bn, o_bn)$. Here,lets consider $(i_an, o_an) and (i_bn, o_bn)$ are the inputs of black-box tests with respect to A and B.
- 5.Now , we will calculate the entropy as mentioned in the problem3.1 by using the formulae: H(x)= -Σ_{i}p$_i$ log_2 p_i.
- 6.We will also calculate the information gain : H(X)- Σ $(|X_v|/|X|)$ $\times$H(X$_v$) , same as mentioned in the above problem's algorithm.
- 7.Now we will have two non deterministic metrics M_a and M_b.

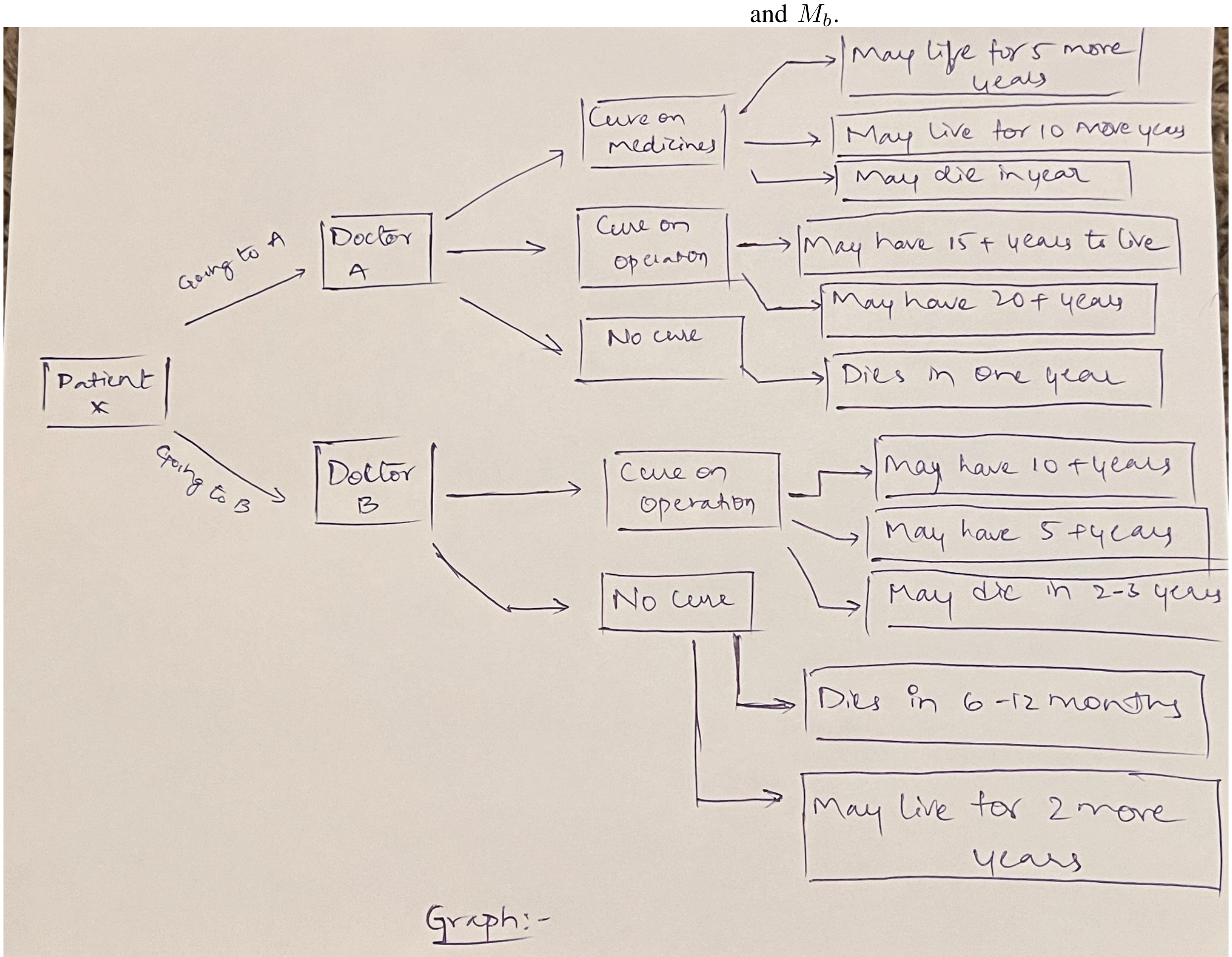


Fig. 1. GRAPH .

IV. ANALYSIS :

If we assume that the metric M_a is bigger than metric M_b. This refers that the metric for case A (M_a) will consists of more output choices(non deterministic choices) than that of the metric for case B(M_b). We can also say that for blackbox case A's output will more likely to shift with the same number of stationary points while the blackbox case B's tends to be stable.

V. STRENGTHS :

There is no need for normalization in this methodology.Even with the values not being there it does not concerns the process and it can still evaluate upto some extent.It helps us to determine non linear relationships that is essential for classifying non-linearly separable data. This is a non parametric approach because we does not have any assumptions regarding the spatial distribution and the structure.

VI. WEAKNESSES :

There are not many more weaknesses for this methodology but we will face some challenges in this. Even with a little change in the data will leads to a big change in the whole structure which will lead to instability.This method takes time. This is complex. The more the test cases or criteria's the complex it will become to manage.

VII. CONCLUSION :

Here we have employed the algorithm and stated a application where the algorithm can be applied in the real life context. We have took the help of entropy to determine a metric which can be used to measure the non determinism under a certain test case. Here we have also disclosed about some of the advantages and disadvantages associated with them.

VIII. REFERENCES:

https://www.mdpi.com/1099-4300/22/4/478
https://codecrucks.com/deterministic-and-non-deterministic-algorithms/
https://www.statisticshowto.com/deterministic-function-nondeterministic/
https://ieeexplore.ieee.org/document/8691615
https://www.cs.virginia.edu/ mohammad/files/papers/BlackBoxes.pdf
https://www.sciencedirect.com/science/article/pii/S0950584923000277